

## artdaq Utilities - Bug #23260

### When creating boot.fcl, DAQInterface should be able to handle default process values

09/11/2019 02:25 PM - John Freeman

<b>Status:</b>	Reviewed	<b>Start date:</b>	09/11/2019
<b>Priority:</b>	Normal	<b>Due date:</b>	
<b>Assignee:</b>		<b>% Done:</b>	100%
<b>Category:</b>		<b>Estimated time:</b>	0.00 hour
<b>Target version:</b>		<b>Spent time:</b>	0.00 hour
<b>Experiment:</b>	-	<b>Co-Assignees:</b>	
<b>Description</b>			
<p>The artdaq processes launched by DAQInterface have default values for port (partition-dependent) and subsystem (1), which can be overridden in the file passed on the boot transition (typically called "boot.txt"). E.g., say we have a parent subsystem, subsystem 1, and a child subsystem, subsystem 2, and they each contain an eventbuilder. Then we could have the following:</p>			
<pre>EventBuilder host: localhost EventBuilder label: ParentSubsystemEventBuilder  EventBuilder host: localhost EventBuilder label: ChildSubsystemEventBuilder EventBuilder subsystem: 2</pre>			
<p>What's important to note is that there's a variable number of lines defining each process in boot.txt. <b>Currently</b>, when FHiCL-izing boot.txt, DAQInterface takes each value type and puts it in a list, e.g.:</p>			
<pre>EventBuilder_hosts: ["localhost", "localhost"] EventBuilder_labels: ["ParentSubsystemEventBuilder", "ChildSubsystemEventBuilder"]</pre>			
<p>...where the idea is the first entry in EventBuilder_labels refers to the same process as the first entry in EventBuilder_hosts, etc. However, the algorithm's not sophisticated enough to deal with default values, meaning that unless every process has its potentially-default value set explicitly, there's no longer a coherent mapping across the lists. What we want is something like the following:</p>			
<pre>EventBuilder_subsystems: ["default", "2"]</pre>			
<p>...where then in DataflowConfiguration.fcl, also saved in the database run record, we could actually save the true value:</p>			
<pre>EventBuilder_subsystems: ["1", "2"]</pre>			

#### History

##### #1 - 09/12/2019 12:02 PM - John Freeman

Kurt pointed out an even better option - because it means you wouldn't explicitly need to write "default" to denote that there was no value in the boot file - would be to give each process its own table. To take the example above, this would become:

```
[ { name: "EventBuilder" host: "localhost" label: "ParentSubsystemEventBuilder" }, { name: "EventBuilder" host: "localhost" label: "ChildSubsystemEventBuilder" subsystem: 2 } ]
```

##### #2 - 09/13/2019 01:54 PM - John Freeman

- % Done changed from 0 to 100

- Status changed from New to Resolved

With commit c61aad420dbc5c6ce5fb4e5041689e8be998f05 at the head of bugfix/issue23260\_handle\_default\_values\_in\_database, I'm marking this issue as resolved. As can be seen in, e.g., the archived run record in the database for run 1001814 on the ProtoDUNE cluster, the following are now available:

- For boot.fcl, two lists of tables, referred to as subsystem\_settings and artdaq\_process\_settings, which describe how subsystems and artdaq processes have been defined in the boot file
- For DataflowConfiguration.fcl two lists of tables, referred to as subsystem\_values and artdaq\_process\_values. subsystem\_values is the same as boot.fcl's subsystem\_settings, though with the different context implied by the change in the name. However, artdaq\_process\_values is actually a superset of the contents of artdaq\_process\_settings, since it refers to not only the artdaq process values which were explicitly set in the boot file, but others which either went to default (e.g., port or subsystem) or aren't ever set in the boot file (e.g., rank)

### #3 - 09/18/2019 01:57 PM - Kurt Biery

I have successfully verified the changes associated with this Issue by running tests at protoDUNE with, and without, the changes and comparing the results.

- run 9662 used DAQInterface v3\_00\_07c and a system that used the DFO (i.e. with two artdaq subsystems)
- run 9663 used DAQInterface v3\_00\_07c and a system that used the RM (i.e. with one artdaq subsystem)
- run 9664 used the DAQInterface code associated with this Issue and a system that used the RM (i.e. with one artdaq subsystem)
- run 9665 used the DAQInterface code associated with this Issue and a system that used the DFO (i.e. with two artdaq subsystems)

The expected differences were seen when comparing the boot.fcl and DataflowConfiguration.fcl files that were stored in the database between runs 9665/9662, 9664/9663, and 9665/9664.

I'm not sure that this test fully exercised the functionality that stores the actual values that were used in the DAQ in the DataflowConfiguration.fcl file (since the JCOP RunControl tends to populate all of the values in boot.fcl). However, the presence of the "rank" parameters in DataflowConfiguration.fcl gives me confidence that this aspect of the new code is working just fine.

I will merge the branch to "develop" in a few moments, and I'll update the "artdaq branches" wiki page.

### #4 - 09/18/2019 02:42 PM - Kurt Biery

- Status changed from Resolved to Reviewed